(1) The user dials into the dial-up server. The server authenticates the user based on any one of the available mechanisms in the module.

(2) The dial-up server invokes the Kerberos client process and uses the user identifier and password to authenticate the user to the network.

(3) If Kerberos authentication is successful, user access to network elements will proceed with the security services offered by the Kerberos network security servers.

(4) If Kerberos authentication times out due to problems in the network or in the Kerberos network security servers, after a number of repeated such failures, the dial-up server will switch user to the dial-up access network and proceed with the steps that are used for user access to the network elements.

There are two issues that need to be addressed for the automated switching of user dial-up access to network elements. First, the number of failures that are allowed for Kerberos network authentication before the user is switched automatically to the dial-up access network shall be determined and set up by the system security administrator. Regular users shall not be allowed to specify and to change this setup. The time-out value for the determination of Kerberos network authentication failure shall also be limited to the system security administrator. Note that the failure of user Kerberos network authentication and that of the network or Kerberos network security servers should be differentiated. The first type of user authentication failure is caused by the user not being able to be authenticated and, therefore, shall be treated as a normal response as far as the dial-up server is concerned. In this case, the user will be notified of the authentication result and shall not be switched over to the dial-up access network. To the dial-up server, only the user authentication failures that are caused by no response which would eventually triggers the time-out or by some error response conditions that clearly indicate the network or Kerberos failure count towards the final determination to automatically switch the user to the dial-up access network.

Second, a dial-up user needs to first dial and connect to the dial-up server before further network authentication takes place in the order of through the Kerberos network security mechanisms then through the dial-up access network in the case of unavailability of the network or the Kerberos network security services. The procedure to dial into the dial-up server depends on the way it is actually implemented.

The dial-up server can be deployed where it is close to the users. Since it is merely a client to the network security servers, the dial-up server **1002** does not have to be a powerful server machine. Therefore, the cost of a large scaled deployment should not be very high. The benefit, on the other hand, is that this would result in a dramatically increase in the number of network access points of presence, even in places where it is not currently feasible to use dial-up access that is based on the toll-free numbers.

5. Computer Program Product

An exemplary computer environment for implementing one or more of the servers, user elements or network elements according to the invention is shown in FIG. **9**. The environment is a computer system **900** that includes one or more processors (CPU), such as processor **904**. The processor **904** is connected to a communications bus **906**. Various software embodiments are described in terms of this example computer system. After reading this description, it will be apparent to a person skilled in the relevant art how to implement the invention using other computer systems and/or computer architectures. Computer system **900** can be used to implement PC **104** and/or the PBX **114**.

Computer system **900** also includes a main memory **908**, preferably random access memory (RAM), and can also include a secondary memory **910**. The secondary memory **910** can include, for example, a hard disk drive **912** and/or a removable storage drive **914**, representing a floppy disk drive, a magnetic tape drive, an optical disk drive, etc. The removable storage drive **914** reads from and/or writes to a removable storage unit **918** in a well known manner. Removable storage unit **918** represents a floppy disk, magnetic tape, optical disk, etc., which is read by and written to by removable storage drive **914**. As will be appreciated, the removable storage unit **918** includes a computer usable storage medium having stored therein computer software and/or data.

In alternative embodiments, secondary memory **910** may include other similar means for allowing computer programs or other instructions to be loaded into computer system **900**. Such means can include, for example, a removable storage unit **922** and an interface **920**. Examples can include a program cartridge and cartridge interface, a removable memory chip (such as an EPROM, PROM, or PCMCIA) and associated socket, and other removable storage units **922** and interfaces **920** which allow software and data to be transferred from the removable storage unit **922** to computer system **900**. Any of the aforementioned memory devices can be used to implement a database.

Computer system **900** can also include a communications interface **924**. Communications interface **924** allows software and data to be transferred between computer system **900** and external devices via communications path **926**. Examples of communications interface **924** can include modem **106**, printer **108**, a communications port, etc. Software and data transferred via communications interface **924** are in the form of signals that can be electronic, electromagnetic, optical or other signals capable of being received by communications interface **924** via communications path **926**. Note that communications interface **924** provides a means by which computer system **900** can interface to a network such as LAN **110**.

The present invention is preferably implemented using software running (that is, executing) in an environment similar to that described above with respect to FIG. **9**. Thus, the term "computer program product" is used to generally refer to a program stored at removable storage device **918** or a hard disk installed in hard disk drive **912**. These computer program products are means for providing software to computer system **900**.

Computer programs (also called computer control logic) are stored in main memory and/or secondary memory **908** and/or **910**, respectively. Computer programs can also be received via communications interface **924**. Such computer programs, when executed, enable the computer system **900** to perform the features of the present invention as discussed herein. In particular, the computer programs, when executed, enable the processor **904** to perform the features of the present invention. Accordingly, such computer programs represent controllers of the computer system **900**.

In an embodiment where the invention is implemented using software, the software may be stored in a computer program product and loaded into computer system **900** using removable storage drive **914**, hard drive **912** or communications interface **924**. Alternatively, the computer program product may be downloaded to computer system **900** over communications path **926**. The control logic (software), when executed by the processor **904**, causes the processor **904** to perform the functions of the invention as described herein.